38   KNOWLEDGE-BASED REQUIREMENTS ENGINEERING FOR RECONFIGURABLE PRECISION ASSEMBLY SYSTEMS

Hitendra Hirani
Precision Manufacture Group
University of Nottingham
epxhjh@nottingham.ac.uk
Svetan Ratchev
Precision Manufacture Group
University of Nottingham
Svetan.ratchev@nottingham.ac.uk
UK

Reconfigurable Precision Assembly Systems are being developed in response to assembly systems becoming obsolete due to condensed product life cycles being so closely linked with assembly system life cycles. However methods and tools to promote the use of the hardware technology, which is centred on the deployment of reconfigurable modules, are non-existent. The paper presents a knowledge-based requirements engineering approach that gathers user requirements and converts them into system requirements using knowledge rules stored within a database structure. This is illustrated through a case study.

1   INTRODUCTION

Reconfigurable Precision Assembly Systems have been highlighted as one of the visionary manufacturing challenges for 2020 (Bollinger, 1998). Although there are initiatives being undertaken that aim to develop hardware for the concept (Mehrabi, 2002; Koren, 1999; Monfared, 1997; Heilala, 2001; Chen, 2001) there are no schemes that aim to develop methods and tools to disseminate the research to industry.

Requirements specification is the key activity in filling this gap as it forms the first stage of the assembly system design process (Bray, 2002). The user defines a set of user requirements and these are converted into system requirements by the system integrator.

The role of requirements engineering is to provide an abstract solution for a design problem. Moreover *"a good set of requirements defines precisely what is wanted, but simultaneously leaves the maximum space for creative design.* (Stevens, 1995) These requirements have to reflect the customer's expectations of the system.

The interaction between the machine and its environment is the key aspect to consider here and all system properties must be defined using these terms. One

method of having concrete requirements to work from is to develop formal models, tools and techniques (Jackson, 1995).

Although there are many commercial tools (International Council on Systems Engineering, 2004; easyweb, 2004) that perform requirements engineering functions, these are mainly tailored to suit software engineering aspects. They chart user requirements declared in natural language and system requirements are defined with reference to these. The tools facilitate functions such as traceability analysis, charting history of requirements and consistency and quality checking, but no knowledge intensive activities are performed.

Knowledge engineering is a key aspect for organisations in the 21st century as the increase in movement of employees between jobs means that knowledge carried by those employees also moves (McCampball, 1999; Bender, 1998). It is in the organisations interest to harness this knowledge within formalised structures where it can be applied to perform some tasks carried out by the workers (Robertson, 2000). For requirements specification this means the semi-automated derivation of system requirements from a set of user requirements.

This paper reports on a research initiative that aims to use assembly system design knowledge to gather user requirements, analyse these requirements and then define system requirements based on these user requirements. This has been implemented through a web-based environment. An overview of the research framework is presented with an outline of the requirements engineering process and the knowledge involved. A case study is included to demonstrate the results.

2   KNOWLEDGE ONTOLOGY FOR REQUIREMENTS ENGINEERING OF RECONFIGURABLE PRECISION ASSEMBLY SYSTEMS

The two areas we are concerned with in this research are those of user requirements specification and system requirements specification, where the user requirements specification consists of a set of business requirements, product definition, part definition, part liaison and other constraints. The system requirements specification comprises the description of assembly tasks and requirements for the specification of assembly modules. The two sets of requirements are owned by two different stakeholders where the user requirements specification is owned by the system user and the system requirements specification is owned by the system integrator. Each is underpinned by the respective stakeholder's knowledge. This knowledge is consolidated by a common knowledge model based on assembly system capabilities. The assembly system capability model will not be explained further in this article as it has been explored in depth in Hirani (2002).

An overview of the ontology is illustrated in Figure 1. The requirements engineering process and a representation of the knowledge are presented separately in this paper to maintain clarity of the boundaries between the knowledge base and the class structure.
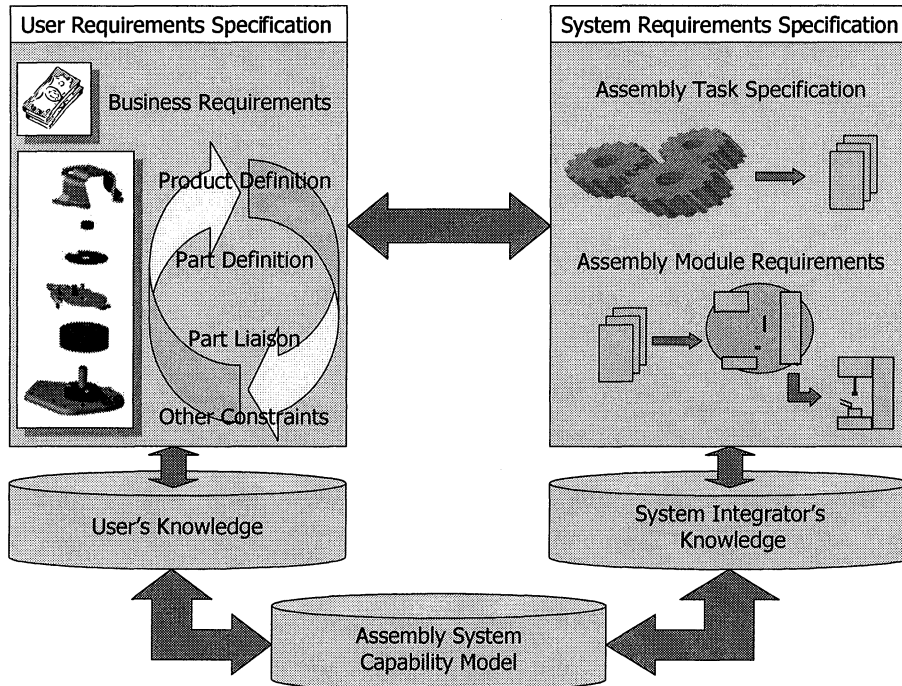
Figure 1: Knowledge Ontology for Requirements Engineering of Reconfigurable Precision Assembly Systems

3 REQUIREMENTS ENGINEERING PROCESS

Requirements engineering for Reconfigurable Precision Assembly Systems begins with the elicitation of user requirements. This entails the specification of business requirements, such as budget, production volumes, delivery timescales, maintenance and training agreements, etc together with a definition of the product(s) to be assembled with their part properties.

A user requirements document is created and sent to the systems integrator, who uses the information to derive a set of task specifications for the finished system to perform. Each task represents the addition of a part to the assembly with handling, feeding and operational properties. These tasks must adhere to the business requirements of the project so they have to be within the universal constraints. All the information derived here is collated to form the system requirements document, which then has to be approved by the system user before any further work is done on the system design. Each item in the system requirements document must be traceable to the user requirement(s) from which it was originally derived. These are properties as illustrated in Figure 2.

For example the Control Architecture is derived from Legacy Systems, Production Volume, Future Modifications, Total Output and System Lifespan user

requirements, whereas Packaging is solely dependent on the Product Delivery required. The knowledge model that underpins the decision making is explored next.
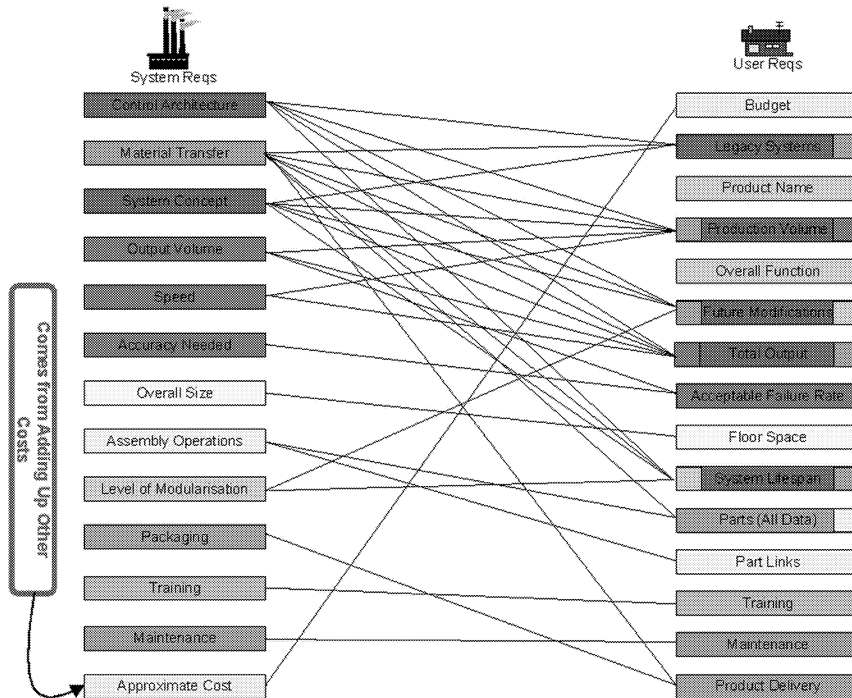

Figure 2: Mapping of System Requirements to User Requirements

4   KNOWLEDGE REPRESENTATION

The knowledge that is needed for requirements engineering for Reconfigurable Precision Assembly System is explored from two viewpoints. The system user's knowledge comprises the knowledge the user has of the business requirements, the products and parts being assembled and the type of liaison that exist between the various parts whilst the system integrator's knowledge includes cost knowledge and technical knowledge on control architectures, material transfer methods and assembly and test operations (see Figure 3).

Knowledge is contained within a database structure at tow levels of abstraction so that it can be easily stored, retrieved and edited as new knowledge is created. Domain knowledge is static knowledge about system properties whereas task knowledge defines the activities that need to be performed by each stakeholder. The link between the domain and task knowledge is facilitated by inference knowledge. This layer describes how domain knowledge should be manipulated for each task and is made up of a series of if then commands for each task. An example of an inference rule is presented in Figure 4.
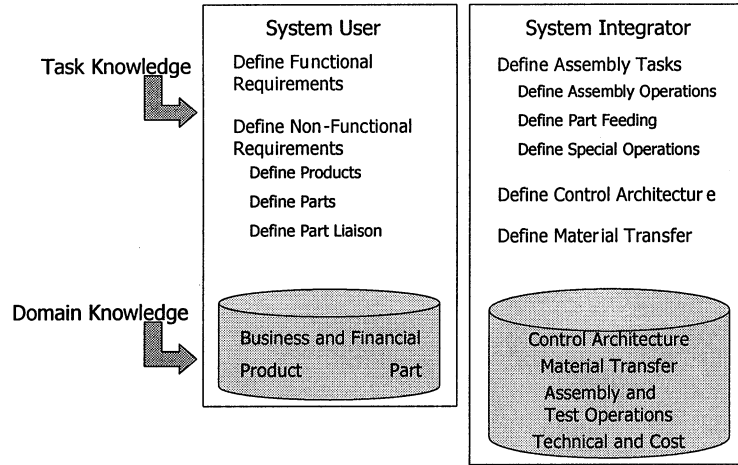
Figure 3: Summary of Requirements Engineering Knowledge

```
;;**************************************************
;;
;;********** CHOOSE OPERATION *********************
;;
(defrule repeatability
        (cycletime ?c&:(>= ?c 4))
        (partlink ?pl)
        =>
        (if (= ?pl screwed) then (assert (operationtype screwing))
        else
         (if (= ?pl joind) then (assert (operationtype joining)))
        )
)
(defrule move
        (movement ?m)
        =>
        (if (= ?m simple) then
                (assert (machines pn_cylinders))
                (assert (machines linear_drives))
        else
                (assert (machines artic_robot))
        )
)
```

Figure 4: Inference Rule for Choosing Operation

The various forms of knowledge are used to define the type of assembly system required and this information is later used to select physical assembly modules that comply with the requirements for assembly of the product. This has been implemented through a web-based decision making environment (Figure 5).

The environment interfaces with the web through a server, which is behind a firewall for security reasons. The server exchanges messages and code with a JSP Servlet which calls different tasks and inferences represented as Java objects and Java beans. This is backed up by a relational database management system that stores the domain knowledge in the MySQL format. The requirements specification stage of the process has been implemented as a prototype environment. It includes the specification of the business constraints for the project, gathering of product

data, data on the parts that make up the product and their connectivity. An example is used to demonstrate the implementation.
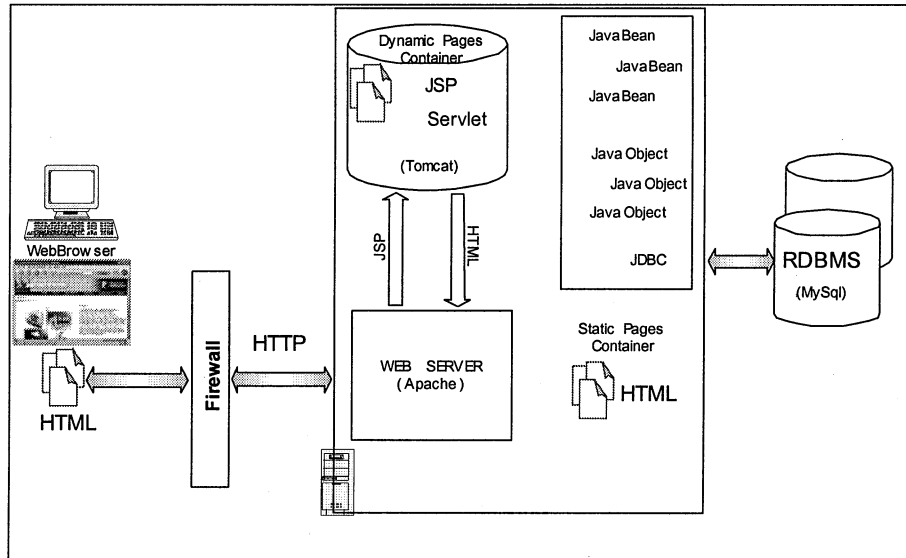

Figure 5: System Architecture for Web-Based Decision-Making Environment

5 CASE STUDY

The requirements specification of a seat recliner mechanism for a car has been performed to demonstrate the use of the system. User requirements for the product assembly have been captured and parsed to form system requirements.

Each part has been described in detail as per the criteria shown in Figure 6 and part liaison characteristics have been declared. These have then been parsed through the system to arrive at a set of task specifications based on inference rules defined within the environment. The result is a set of task descriptions as illustrated in Figure 7.

Each assembly task contains a similar description that can later be used by system integrators to design reconfigurable precision assembly modules that satisfy both the task requirements and the non-functional requirements. These modules would then be integrated to form a Reconfigurable Precision Assembly System.
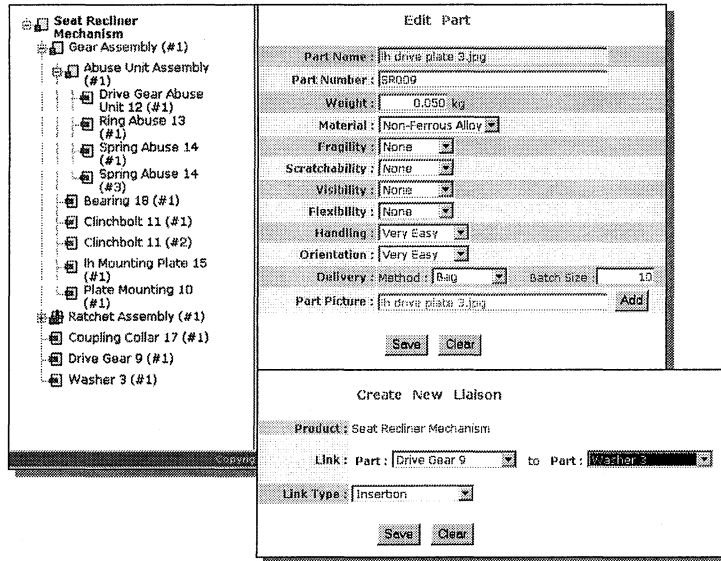
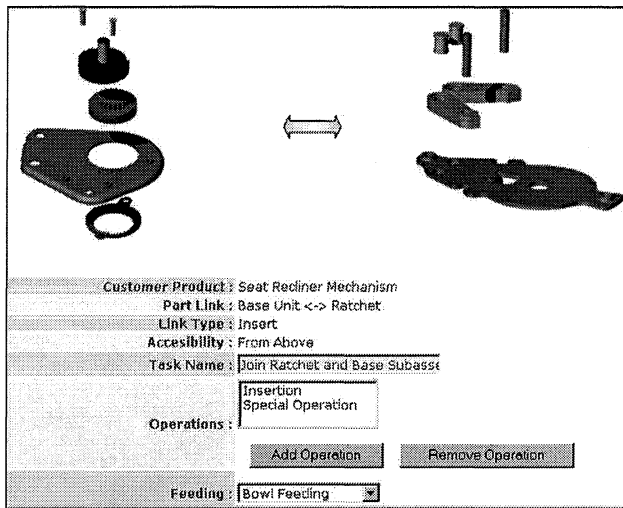Figure 6: Product and Part Descriptions for Seat Recliner Mechanism

Figure 7: Task Specification for Seat Recliner Mechanism

6 CONCLUSIONS

The paper has presented a framework for the knowledge-based requirements specification of a Reconfigurable Precision Assembly System. This approach centres on users supplying knowledge about the product to be assembled and its constituent parts and how they are related together with some business requirements. These are

then parsed through the system using knowledge supplied by system integrators to develop task specifications that satisfy the user requirements.

The method has been implemented through a web-based environment and demonstrated using an example of a seat recliner mechanism in a car. Further work needs to be carried out to expand the knowledge base and to add more functionality to the software and make it holistic.

7   ACKNOWLEDGEMENTS

8   REFERENCES

1.   Bender, S.; Fish, A. (1998): The Transfer of Knowledge and the Retention of Expertise: The Continuing   Need for Global Assignments, Journal of Knowledge Management 4:2, pp.125-137
2.   Bollinger, J. (1998): Visionary Manufacturing Challenges for 2020, National Research Council Publication, National Academy Press, Washington, D.C. (USA)
3.   Bray, I.K. (2002): An Introduction to Requirements Engineering, Pearson Education Ltd, Harlow, Essex (UK) ISBN 0201 767929
4.   I-Ming Chen (2001): Rapid Response Manufacturing Through a Rapidly Reconfigurable Robotic Cell, Robotics and Computer Integrated Manufacturing, 17, pp199-213
5.   http://easyweb.easynet.co.uk/~iany/other/vendors.htm, 6th April 2004
6.   Heilala, J.; Voho, P. (2001): Modular Reconfigurable Flexible Final Assembly Systems, Assembly Automation, 21, 1, pp20-28
7.   Hirani, H.; Ratchev, S. (2002): Definitions and Measures for Requirements Engineering of Reconfigurable Precision Assembly Systems, Proceedings of the 31st International Symposium on Robotics, 7th-11th October, Stockholm, Sweden
8.   http://www.incose.org/tools/tooltax.html, International Council on Systems Engineering, 6th April 2004
9.   Jackson, R.B; Embley, D.W.; Woodfield, S.N. (1995): Developing Formal Object Oriented Requirements Specifications: A Model, Tool and Technique, Information Systems, 20:4, pp273-289
10.   Koren, Y., Heisel, U., Jovane, F., Moriwaki, T., Pritschow, G., Ulsoy, G., Van Brussel, H., (1999) Reconfigurable Manufacturing Systems, Annals of the CIRP, 48:2, pp.527-539
11.   McCampbell, A.S.; Clare, L.M.; Gitters S.H. (1999): Knowledge Management, The New Challenge for the 21st Century, Journal of Knowledge Management 3:3, pp.172-179
12.   Mehrabi M.G; Ulsoy, A.G.; Koren, Y.; Heytler, P. (2002): Trends and Perspectives in Flexible and Reconfigurable Manufacturing Systems, Journal of Intelligent Manufacturing, 13, pp135-146, Kluwer Academic Publishers
13.   Monfared, R.P.; Weston, R.H. (1997): The re-engineering and reconfiguration of manufacturing cell control systems and reuse of their components, proceeds of the institution of mechanical engineers conference, 211:B, pp495-508
14.   Robertson, M,; Hammersley, G. (2000): Knowledge Management Practices Within a Knowledge Intensive Firm: The Significance of the People Management Dimension, Journal of European Industrial Training, 24/2/3/4/ pp.241-253
15.   Stevens, R.; Martin, J. (1995): What is Requirements Management? Proceedings of the 5th Annual International Symposium of the NCOSE, vol 2, pp13-18